

Github esteka:

<https://github.com/alberdijon/Web-garapena-zerbitzari-ingurunean-Azterketa-Jon-Alberdi>

## 2.Ariketa:

Irudian "PROYECTO\_GASTOS" izeneko djangoko proiektu bat ikuzi ahal daiteke. Bere barruan Aplikazio bat sortuta dago gastos izenarekin.

Manage.py: proiektuaren adiministrazioako zereginak egiteko erabiltzen da, adibidez migrazioak egin, erabiltzaileak sortu edo servidorea martxan jarri.

settings.py: proiektuaren konfiguraizoz arduratzen da, bertan proiektuan instalatu beharreko aplikazioak edo datu-datubasearen konexioa egiten da.

models.py: Aplikazioan erabiliko ditugun modeloak edo klaseak definiitzeko erabiltzen da, gainera, bertan eginiko klaseak datu-basean taulak izango dira.

views.py: erabiltzailearen ezkaerak kudeatzeko egiten da, bertan egiten da aplikazioaren logika eta datu-basea eta erabiltzaileak ikusten duena konektatzen du.

urls.py: gure web-horriak izango dituen ezteka ezberdinak kudeatzeko erabiltzen da.

## 3.Ariketa:

Hurrengo akatsa, datubasearekin zer ikusia duen fitxategia ikutu ostean migraziorik egin ez dugula dio. Beraz, models.py edo admin.py fitxategiak aldatetarenbat izan dute baina ez da migrazioa egin. Hau konpontzeko hurrengo 2 kode lerroak idatzi behar dira:

```
python manage.py makemigrations AplikazioarenIzena
```

```
python manage.py migrate AplikazioarenIzena
```

## 4.Ariketa:

Kode hori .html fitxategietan aurkitu genezake. Bere erabillera beste html. batzuk hedatzea da, kasu honetan blog karpeta barruan dagoen list.html fitxategia hedatzen du, beraz, fitxategi horretan dagoen kode dana erakutsiko du berriz idatzi gabe.

## 5.Ariketa:

MVC edo Model View Controller egitura, gure proiektuen egituratzeko modu bat da, bertan datu basearekin egiten den konexioa, erabiltzaileari erakusten zaiona eta aplikazioak bien hartean egiten duen logika

fitxategi ezberdinetan banatzen da.

Djangoren kasuan Model: models.py eta admin.py View: template barruan dauden .html fitxategiak eta Controller: urls.py, views.py eta forms.py.